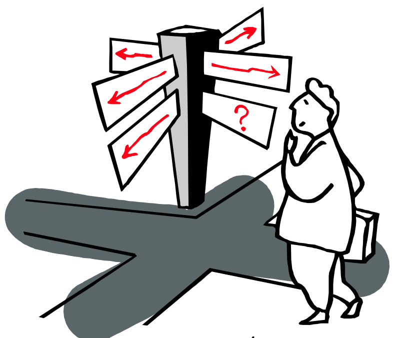


البرمجة بلغة

C++ Programming





1
C++ Programming (1)

المكونات الأساسية للغة C++ وأدواتها

Basic Elements Of C++

Chapter 1

2
C++ Programming (1)

محتويات الفصل الأول

- مقدمة
  - اللغات البنيوية
  - اللغات غرضية التوجه
- بنية البرنامج في لغة C++
  - التعليقات
  - موجهات المهيئ
  - التابع main( )
  - تعليمة الاخراج
  - أمثلة



3
C++ Programming (1)

  - متواليات الهروب
  - التعليمة return 0
  - تعليمة الادخال
  - التعليمة endl
  - أمثلة
  - الأسئلة
- رموز لغة C++
- أنواع المعطيات الأساسية في لغة C++
  - المتحولات المحرفية char
  - المتحولات الصحيحة
  - المتحولات الحقيقية
  - الأسئلة

- العوامل الحسابية
- العوامل العلائقية والمنطقية
- المعامل ثلاثي الحدود
- أولوية العمليات الحسابية
- أمثلة
- الأسئلة

مقدمة

- تعتبر لغة C++ من أشهر اللغات التي تتمتع بطابع القوة والمرونة لإنتاج أسرع برامج وأفضلها أداءاً.
- وعلى الرغم من وجود العديد من لغات البرمجة الأخرى إلا أنها تفتقر شمولية لغة C++ وقوتها .
- فاللغة C++ تتميز بقابليتها على معالجة التطبيقات الكبيرة والمعقدة، والقوة في صيانة البرامج المكتوبة بها مما يوفر وقتاً في تصميم البرامج وتطويرها .
- تعتبر اللغة C++ امتداداً للغة C. وقد أنشأها Bjarne Stroustrup عام 1979 م، وكانت تسمى حينها C مع صفوف (C with classes)، وتغير اسمها إلى C++ في العام 1983م.

6
C++ Programming (1)

- تعتمد اللغة ++C أسلوب البرمجة غرضية التوجه Object Oriented Programming، والذي يعرف اختصاراً بـ (OOP)، والذي تم تطويره بسبب قيود كانت أساليب البرمجة القديمة المتمثلة في اللغات البنيوية تفرضها على المبرمجين.
- ولكي نتعرف على طبيعة تلك القيود يجب أن نلقى الضوء على ما يحدث في اللغات البنيوية.

7 C++ Programming (1)

اللغات البنيوية

- لغات Pascal، C، Basic و Fortran وغيرها من لغات البرمجة التقليدية هي لغات بنيوية أو إجرائية (Procedural).
- أي أن كل عبارة في اللغة هي عبارة عن تعليمة للحاسوب أن ينفذ شيئاً ما : للحصول على دخل أو جمع أرقام الخ.. ... .
- لذا نجد أن البرنامج المكتوب بلغة بنيوية هو عبارة عن لائحة من التعليمات.
- لا تبدو هنالك مشكلة مع البرامج الإجرائية الصغيرة، فالمبرمج ينشئ لائحة التعليمات ويقوم الحاسوب بتنفيذها.
- ولكن مع كبر حجم البرامج لا تعود لائحة من التعليمات فعالة حيث يصعب فهم برنامج يتألف من مئات من العبارات .

8 C++ Programming (1)

- لـذا تـم اعتمـاد أسـلوب التوابـع (Functions) والإجـراءات (Procedures) كوسيلة لجعل البـرامج أسـهل للقراءة والفهم، حيث يمتلك كل تابع في البرنامج واجهة محددة، وينفذ هدفاً محدداً.
- ولكـن المشكلة مـا تـزال قائمـة : مجموعـة مـن التعليمـات تنفذ مهامـاً محددة.
- و مع تزايد حجم البرامج وتعقيدها، يظهر ضعف الأسلوب الإجرائي، حيث تصبح البرامج الضخمة معقدة إلى حد كبير .
- غالبـاً مـا يكـون تصـميم البـرامج الإجرائيـة صـعباً، لأن مكوناتهـا الرئيسية (التوابع) عبارة عن بنية معطيات لا تقلد العالم الحقيقي جيداً. و يصعب في اللغات الإجرائية إنشـاء أي نـوع معطيـات جديد بخـلاف الأنواع المعرفة أصلاً في تلك اللغات ، لكل هذه الأسباب تم تطوير لغات غرضية التوجه .

9
C++ Programming (1)

اللغات غرضية التوجه

- الفكـرة الأساسية وراء اللغـات غرضـية التوجـه هـي دمـج المعطيـات والتوابـع التـي تعمـل علـى تلـك المعطيـات فـي كينونـة واحـدة تسـمى غـرض (Object)، وعـادة تـزود توابـع الغـرض والتـي تسمى توابـع الأعضـاء (Member functions)- الطريقـة الوحيدة للوصـول إلـى المعطيات، لذا تكون المعطيات محمية من التعديلات الخطأ ويقال أن المعطيات وتوابعها مغلقة (Encapsulated) في كينونة واحدة.

10
C++ Programming (1)

بنية البرنامج في لغة ++C

- بنية البرنامج

```
//This program will display a message on the screen.
#include<iostream.h>
void main ( )
{
    cout <<"welcome to C++ Programming ! “  ;
}//End main
```

RESULT
welcome to C++ Programming !

11
C++ Programming (1)

التعليقات:Comments

//This program will display a message on the screen.

- يبدأ هذا السطر من البرنامج بالشرطة المزدوجة (//) الدالة على أن بقية السطر عبارة عن تعليق (comment)، تضاف التعليقات إلى البرامج لتساعد المبرمج أو أي شخص آخر قد يحتاج إلى قراءة البرنامج على فهم ما الذي يفعله البرنامج، لذا من المستحسن أن يبدأ كل برنامج في لغة ++C بتعليق يوضح الغرض الذي من أجله كتب البرنامج.
- تستخدم الشرطة المزدوجة (//) إذا كان التعليق يمتد لسطر واحد فقط single-line comment.هنالك نوع آخر من التعليقات يتيح لنا كتابة تعليقات تمتد إلى عدة أسطر multi-line comments ، نستطيع كتابة التعليق السابق كما يلي:

12
C++ Programming (1)

/*

This program will display a message on the screen

*/

- يبدأ الرمز" */ " ثم التعليق وينهتي بالرمز " /*" . نجد أن نهاية السطر لا تعنى انتهاء التعليق لذا يمكننا كتابة ما نشاء من أسطر التعليقات قبل الانتهاء بالرمز " /* " .
-

13
C++ Programming (1)

موجهات المهيئ (Preprocessor Directive)

#include<iostream.h>

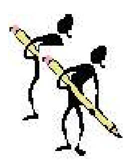
- يسمى هذا موجه المهيئ Preprocessor directive، وهو عبارة عن تعليمة للمترجم أن يدرج كل النص الموجود في الملف iostream.h في البرنامج، وهو ملف يجب تضمينه مع أي برنامج يحتوى على عبارات تطبع معطيات على الشاشة أو تستقبل معطيات من لوحة المفاتيح.
- يسمى iostreamالملف الرأسي (header file)، وهنالك الكثير من ملفات الرأسية الأخرى، فمثلاً إذا كنا نستعمل في برنامجنا توابع رياضية كـ sin( ) وcos( ) نحتاج إلى شمل ملف رأسي يدعى math.h، وإذا كنا نتعامل مع سلاسل الأحرف سنحتاج للملف string.h.

14
C++ Programming (1)

- وعموماً هنالك عدد كبير من الملفات الرأسية التي يجب تضمينها وذلك حسب طبيعة البرنامج .
- تعتبر الملفات الرأسية جزء مهم من برامج لغة C++ وسنحتاج إلى شمل الملف iostream.h لتشغيل أي برنامج يقوم بعمليات الإدخال والإخراج.

C++ Programming (1)
15

التابع main( )



- يبدأ تشغيل أي برنامج C++ من تابع يدعى main( )، وهو تابع مستقل ينقل نظام التشغيل التحكم إليه. وهو جزء أساسي في برنامج C++.
- الأقواس بعد main تشير إلى أن main عبارة عن تابع.
- قد يحتوى برنامج C++ على أكثر من تابع أحدهما بالضرورة هو main, ويحتوى البرنامج السابق على تابع واحد.
- يبدأ تنفيذ البرنامج من التابع main حتى لو لم يكن هو الأول في سياق البرنامج , ويتم حصر جسم التابعmain بأقواس حاصرة { } .

C++ Programming (1)
16

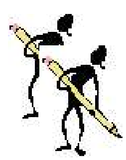
تعليمة الخرج على الشاشة

cout <<"welcome to C++ Programming ! “ ;

- هذه التعليمة (statement) تجبر الحاسوب أن يظهر على الشاشة النص المحصور بين علامتي الاقتباس " ". ويسمى هذا النص بسلسلة محرفية.
- يجب أن تنتهي كل تعليمة في برنامج C++ بفاصلة منقوطة ; (semi colon).
- الكلمة المحجوزة أو العبارة cout والذي تلفظ كـ C out يمثل غرض في C++ مقترن مع الشاشة والعامل >> والذي يسمى بعامل الوضع Put to operator يجبر على إرسال الأشياء التي على يمينه إلى أي شئ يظهر على يساره.

17 C++ Programming (1)

مثال



```
#include <iostream.h>
void main ( )
{
   cout  << 7 << " is an integer.\n";
   cout << 'a' << "is a character.\n";
}
```

RESULT
7 is an integer.
a is a character

18 C++ Programming (1)

- **من خرج البرنامج يتضح لنا ما يلي:**

1- يتم حصر النص المطلوب ظهوره على الشاشة بين علامتي اقتباس مزدوجة " is an integer ".

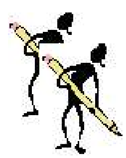
2- تتم كتابة الثوابت الرقمية بدون علامتي اقتباس cout<< 7.

3- يتم حصر حرف واحد مطلوب ظهوره على الشاشة بعلامة اقتباس فردية cout << ' a '.

تقوم بعض اللغات كـ Basic مثلاً بالانتقال إلى سطر جديد تلقائياً في نهاية كل عبارة خرج ، لكن C++ لا تفعل ذلك كما أن العبارات المختلفة والموضوعة في أسطر مختلفة لا تؤدي إلى ذلك .

لا تنشئ العبارة cout أسطراً جديدة تلقائياً، والمخرجات في البرنامج التالي توضح ذلك:

19
C++ Programming (1)



مثال

```
#include<iostream.h>
void main ( )
{
 cout<<10;
 cout<<20<<30;

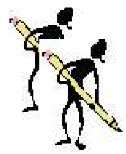
}
```

RESULT

102030

20
C++ Programming (1)

- حيث يلتصق كل الخرج ببعضه البعض ، لذا من الأفضل أن يكون لدينا طرق في C++ للتحكم بطريقة تنسيق الخرج والتي منها متواليات الهروب (Escape Sequences).



21 C++ Programming (1)

متواليات الهروب (Escape Sequences)

- نلاحظ أنه لم تتم طباعة \n على الشاشة ، \ تسمى الشرطة الخلفية (Back) slash أو حرف هروب (Escape character) وتسمى هي والحرف الذي يليها متواليات الهروب.
- متواليات الهروب \n يعنى الانتقال إلى سطر جديد حيث يجبر المؤشر على الانتقال إلى بداية السطر التالي ، نعرض بعض متواليات الهروب الشائعة .

22 C++ Programming (1)

• متوالية الهروب   الوصف

- \n   سطر جديد.
- \t   مسافة أفقية.
- \b   حرف التراجع back space.
- \\   لطباعة شرطة خلفية.
- \r   حرف الإرجاع، يجبر المؤشر على الانتقال إلى بداية هذا السطر.
- \"   لطباعة علامة اقتباس

23
C++ Programming (1)

التعليمة   return 0

- تكتب العبارة ;return 0 في نهاية التابع ( )main إذا لم يسبقه العبارة void.
- القيمة 0 تشير إلى أن البرنامج انتهى نهاية صحيحة وسيبدو لنا سبب تضمين هذه العبارة واضحا عندما نتعرف على التوابع في C++ بالتفصيل.

```
#include<iostream.h>
int main ( )  {
            ……………
return 0;
}
```


24
C++ Programming (1)

تعليمة الادخال من لوحة المفاتيح

cin>>integer1 ;

- هذه العبارة تخزن الرقم الذي يكتبه المستخدم من لوحة المفاتيح في متغير يدعي integer1 .
- تمثل العبارة cin (والذي تلفظ كـ C in) الادخال من لوحة المفاتيح، ويأخذ عامل الحصول (>>) get from الأشياء الموضوعة على يساره ويضعها في المتغير الموجود على يمينه.
- وعند تنفيذ هذه العبارة ينتظر البرنامج أن يكتب المستخدم رقماً من النوع integer ويضغط على مفتاح Enter، ويتم إسناد القيمة التي أدخلها المستخدم إلى المتغير integer1 .

- يمكن استعمال عامل الادخال عدة مرات في نفس العبارة:

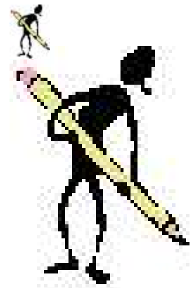
```
cin >> integer1>>integer2;
```

- يضغط المستخدم هنا Enter، أو مفتاح المسافة Space، أو مفتاح Tab بعد كل قيمة، قبل أن يكتب القيمة التالية، ولكنه من الأفضل عادة إدخال قيمة واحدة في كل مرة لتجنب الخطأ.
- البرنامج التالي يوضح الدخل بواسطة cin والخرج بواسطة cout.

مثال

اكتب برنامجا لإدخال رقمين من المستخدم ويجمعهما و يطبع ناتج الجمع:

```
// Addition program
#include<iostream.h>
void main ( )  {
    int integer1, integer2, sum;
    cout <<"Enter first integer\n";
    cin >> integer1;
    cout <<"Enter second integer\n";
    cin >> integer2;
    sum= integer1+integer2;
    cout <<"sum="<<sum<<endl;
  }
```



RESULT

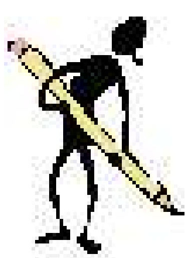
```
Enter first integer
7
Enter second integer
3
sum= 10
```


27

التعليمة endl

cout<<"sum= "<<sum<<endl

- تطبع النص المحرفي =sum متبوعاً بقيمة sum ، ونلاحظ أننا استخدمنا endl وهو وسيلة أخرى فيC++ للانتقال إلى سطر جديد، ويسمى مناور manipulator
- و endl اختصاراً لـ end line، وهو يعمل تماماً كما يعمل متوالية الهروب \n .



C++ Programming (1)

رموز لغة ++C

- يبنى البرنامج في لغة ++C من مجموعة من العناصر التالية :
  - الحروف الانكليزية الصغيرة a, b , c, d, . . . x, y, z
  - الحروف الانكليزية الكبيرة A, B, C, D, . . . X, Y ,Z
  - الأرقام العربية 0, 1, 2, 2, . . . 8, 9
  - رموز خاصة مثل :

< <= > >= + - * / = # % $

? ; : , . << >> ! ‘ ”

{ } [ ] || & ( )

29 C++ Programming (1)

أنواع المعطيات الأساسية في لغة ++C

- هنالك سبعة أنواع معطيات أساسية في ++C ، واحد منها يمثل الأحرف وثلاثة تمثل أعداداً صحيحة وثلاثة تمثل أعداداً حقيقية.و الجدول الآتي يلخص هذه الأنواع.

30 C++ Programming (1)

أمثلة عن القيم المخزنة	يستعمل لتخزين	اسم النوع
‘ a’	أحرف(محارف)	char
222	أعداد صحيحة قصيرة	short
153 406	أعداد صحيحة عادية	int
123 456 789	أعداد صحيحة طويلة	long
3.7	أعداد حقيقية قصيرة	float
7 533 .039 395	أعداد حقيقية مزدوجة	double
9176 321 236.012026	أعداد حقيقية ضخمة	long double

31 C++ Programming (1)

المتحولات Variables

- **المتحول:** هو عملية تحديد موضع في الذاكرة يمكن أن نخزن فيه قيما ما ليتم استخدامها في البرنامج.
- . التصريح عن المتحولات:
- قيمة ابتدائية = اسم المتحول   نمط المعطيات;

int x=200   ;   long y=300 ;

- يمكن التصريح عن المتحولات في أي مكان من البرنامج, ولكن لابد من التصريح عن المتحول قبل استخدامه.
- يمكن تهيئة المتحولات مباشرة أثناء التصريح عنها (إعطاء قيم ابتدائية).
- يمكن التصريح عن أكثر من متحول من نفس النمط بتعليمة واحدة, بحيث يفصل بينها ,

int x1=200,   x2=100   ;

32 C++ Programming (1)

المتحولات Variables

- **يجب إتباع ما يلي عند تسمية المتحولات :**

1. يجب أن يكون اسم المتحول معرفاً صحيحاً , أي سلسلة من الرموز تبدأ بحرف من حروف اللغة الانكليزية .
2. أن لاتبدأ السلسلة برقم أو (_) .
3. يجب أن لا يكون كلمة محجوزة مثل for , أو أي قيمة منطقية true , false , او الكلمة null .
4. يجب ألا يحتوي الاسم فراغات, أو رموز اللغة الخاصة (+,-,@,#...)
5. يفضل أن يبدأ اسم المتحول بحرف صغير , وفي حال كان الاسم مؤلف من أكثر من كلمة يفضل جعل الحرف الأول من كل كلمة (ما عدا أول كلمة) حرفاً كبيراً مثل : isGoodColor
6. يمكن أن يصل طوله حتى (31) محرف.
7. اختيار أسماء معبرة للمتحولات.

33
C++ Programming (1)

المتحولات المحرفية char

- يتم تخزين الأحرف(المحارف) في متحولات من النوع char, والعبارة التالية توضح ذلك :

char ch;

- تنشئ مساحة في الذاكرة لمحرف وتسميه ch وحجمه 1Byte
- ولتخزين محرف ما في هذا المتغير نكتب :

ch = ’z’

- ودائماً تكون المحارف الثابتة كـ ‘a’ و ’b’ محصورة بعلامة اقتباس فردية.

34
C++ Programming (1)

- يمكن استعمال المتحولات من النوع char لتخزين أرقام كاملة بدلاً من محارف ، فمثلاً يمكننا كتابة:

```
ch=2;
```

- لكن نطاق القيم الرقمية التي يمكن تخزينها في النوع char يتراوح بين 128– إلى 127 لذا فإن هذه الطريقة تعمل مع الأرقام الصغيرة فقط.

35
C++ Programming (1)

المتحولات الصحيحة

- تمثل المتحولات الصحيحة أرقاماً كاملة أي قيم يمكن تعدادها ، كعدد أشخاص أو أيام أو عدد صفحات مثلاً ، ولا يمكن أن تكون الأعداد الصحيحة أرقاماً ذات نقطة عشرية ولكنها يمكن أن تكون سالبة.
- هنالك ثلاثة أنواع من الأعداد الصحيحة في C++:
  - short قصير
  - int عدد صحيح
  - long طويل
- وهي تحتل مساحات مختلفة في الذاكرة. الجدول التالي يبين هذه الأنواع والمساحة التي تأخذها في الذاكرة ونطاق الأرقام التي يمكن أن تأخذها:

36
C++ Programming (1)

النطاق	الحجم	اسم النوع
128- إلى 127	1byte	char
32,768- إلـــــى 32,767	2byte	Short
مثل short في أنظمة 16bit ومثل long في أنظمة 32bit int حجمه 4Byte		
- 2,147,483,648 إلـــــــــــى 2,147,483,647	8byte	long

37
C++ Programming (1)

المتحولات الحقيقية (float)

- يتم استعمال المتحولات الحقيقية لتمثيل قيم يمكن قياسها كالأطوال أو الأوزان.
- ويتم تمثيل الأعداد الحقيقية عادة برقم كامل على اليسار مع نقطة عشرية وكسر على اليمين.
- هنالك ثلاثة أنواع من الأعداد الحقيقية في أنظمة التشغيل الشائعة الاستعمال,وأشهر أنواع الأعداد الحقيقية من النوع double والذي يتم استعماله لمعظم توابع C++ الرياضية.ويتطلب النوع float ذاكرة أقل من النوع double.
- و يوضح الجدول التالي هذه الأنواع والحجم الذي تأخذه في الذاكرة.

38
C++ Programming (1)

الحجم	اسم النوع
4byte	float
8byte	double
10byte	long double

39
C++ Programming (1)

العوامل الحسابية (Math Operator)

- لقد استعملنا عامل الجمع (+) لجمع integer1 وinteger2 ، تتضمن ++C العوامل الحسابية التالية :
  - هي : - , + , % , / , *
  - الزيادة و النقصان : -- , ++
  - الإدخال والإخراج , >> , <<
  - الاسناد =
  - والمعاملات التالية :
  - -= , += , *= , %= , /=

40
C++ Programming (1)

الجدول التالي يبين العوامل الحسابية الأساسية :

العامل	الوظيفة	التعبير الجبري	التعبير في C++
+	جمع	B+h	B+h
-	طرح	B-h	B-h
*	ضرب	Bh	B*h
/	قسمة	B/h,	B/h
%	باقي القسمة	B mod h	B%h

41
C++ Programming (1)

- العوامل الأربعة الأولى تنجز أعمالاً مألوفة لدينا، أما عامل باقي القسمة % المسمى أيضاً المعامل modulus، يتم استعماله لحساب باقي القسمة لعدد صحيح على عدد آخر، لذلك فالتعبير 20%3 يساوى 2 .
- تسمى هذه العوامل الحسابية بالعوامل الثنائية لأنها تعمل على قيمتين.
- يمكن استعمال أكثر من عامل في تعبير رياضي واحد، فمثلاً التعبير:

```
C=(f-32)*5/9;
```

يحول درجة الحرارة من مئوية إلى فهرنهايت.(استعملت الأقواس لكي يتم تنفيذ الطرح أولاً بالرغم من أولويته المتدنية، يشير المصطلح أولوية Precedence إلى ترتيب تنفيذ العوامل، العاملان * و / لهما أولوية أعلى من +و-).وهذا ما سنراه لاحقاً بعد أن نتعرف على بقية عوامل C++ .

42
C++ Programming (1)

المعاملات العلائقية والمنطقية
Boolean/Relational Operators

- تقارن العوامل العلائقية قيمتين، وتؤدى إلى نتيجة صح/خطأ وفقاً لما إذا كانت المقارنة صح/خطأ.
- هنالك ستة عوامل علائقية مبينة في كما يلي:
- المساواة وعدم المساواة : != , ==
- علاقات : >= , <= , > , <
- علاقة الشرط (AND(Conditional : &&
- علاقة الشرط (OR(Conditional : ||

43 C++ Programming (1)

يبين الجدول التالية العوامل العلائقية أو المقارنة مع مثال لكل نوع

الرمز	المعنى	مثال
==	يساوى	a==b
!=	لا يساوى	a!=b
>	أكبر من	a>b
<	أصغر من	a<b
>=	أكبر من أو يساوى	a>=b
<=	أصغر من أو يساوى	a<=b

44 C++ Programming (1)

- تكون التعابير المبينة في عمود المثال صحيحة أو خطأ وفقا لقيم المتغيرين a و b.
- فلنفرض مثلاً أن:

a يساوى 9 و b يساوى 10.

التعبير : a == b خطأ.

- التعبير a!=b صحيح وكذلك التعبيرين a<b و a<=b ، والتعبيرين a>b و a>=b خطأ..

45
C++ Programming (1)

أولوية العمليات الحسابية

( )

fabs( ),pow( ) , sqrt( ) ,sin( ) ,cos( ) . . .

* / %

+ -

<< , >>

< , > , <= , >=, == , !=

&& , || , !

= , += , -=, *= , /= , %=

46
C++ Programming (1)

1.5 * 2.4 + 3.3 * 4.25 / 5.5

3.6 + 3.3 * 4.25 / 5.5

3.6 + 14.025 / 5.5

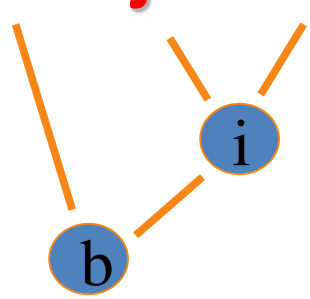
3.6 + 2.55

6.15

47
C++ Programming (1)

Evaluating Expressions

```
int i = 1 , j = 2,  k = 3;
i < j - k
```

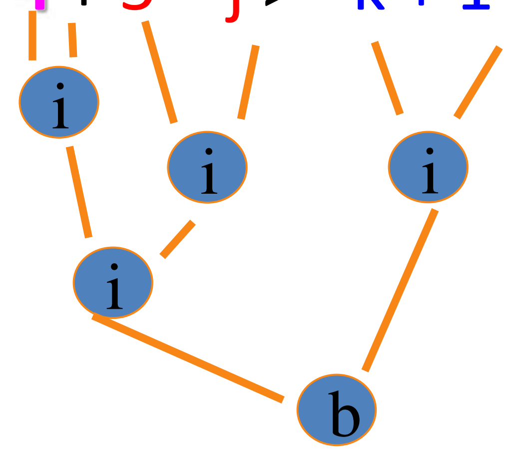


48
C++ Programming (1)

Evaluating Expressions

int i = 1, j = 2, k = 3;

-i + 5 * j >= k + 1



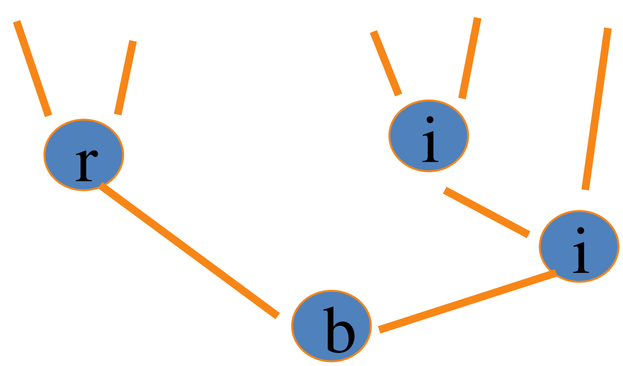
49
C++ Programming (1)

Evaluating Expressions

int i = 1, j = 2, k = 3;
double x = 5.5, y = 7.7

x - y <= j - k + 1

(x - y) <= ((j - k) + 1)



50
C++ Programming (1)

```
// Relational_Operators
#include <iostream.h>
void main ( )
{
    int  m=2, k=3, i=1;
    cout <<"i="<<i<<"\tk="<<k<<"\tm="<<m<<endl;
    cout <<"(i>=1 && 2<4)= "<<( i>=1 && 2<4)<<endl;
    cout <<"(2<=99 && k<m)= "<< (2<=99 && k<m) <<endl;
    cout <<"(2 >=i || k==m)= "<<(2 >=i || k==m) <<endl;
    cout <<"(k+m<2 ||3-2>=k)="<<(k+m<2 ||3-2>=k)<<endl;
    cout <<"(!m)= "<<(!m)<<endl;
    cout <<"(!(2-m))= "<<(!(2-m))<<endl;
    cout <<"(!(k>m))= "<<(!(k>m))<<endl;
}
```


الأسئلة

- **أكتب عبارة C++ صحيحة تقوم بالآتي:**
- تعريف المتغيرات x ،y ، z و result لتكون من النوع int.
- الطلب من المستخدم إدخال ثلاثة أرقام صحيحة.
- **حدد ما إذا كانت العبارات الآتية صحيحة أم خطأ:**
- يجب الإعلان عن المتغيرات قبل استعمالها في البرنامج.
- يجب تحديد نوع المتغيرات عند الإعلان عنها.
- لا تفرق C++ بين المتغيرات Number و number .

52 C++ Programming (1)

- اكتب برنامجاً يطبع على الشاشة أي معلومات شخصية عنك تريدها كـ (اسمك ، رقمك الجامعي ، تاريخ ميلادك ، عنوانك ....) بشكلٍ مُنسق ؛ مستخدماً تتابعات الهروب (\n) وَ (\t).
- برنامجاً يقوم بحساب مربع عدد حقيقي يُدخله المستخدم من لوحة المفاتيح ؛ ثم يطبع النتيجة على الشاشة .
- برنامج لتنفيذ العبارة الرياضية التالية [ ] بصيغ مختلفة باستخدام الأقواس.

y = (6+12/6)*2-1; y = (6+12)/(6*2-1); y = 6+12/6*2-1;

- 15 , 1 , 9

الأسئلة

- **حدد ما إذا كانت العبارات الآتية صحيحة أم خطأ:**
- التعليقات تجبر الحاسوب على طباعة النص الذي يلي // على الشاشة عند تنفيذ البرنامج.
- تتابع الهروب n\ يجبر المؤشر على الانتقال إلى سطر جديد.
- برنامج ++C والذي يقوم بطباعة ثلاث أسطر على الشاشة يجب أن يحتوى على ثلاث عبارات تستعمل cout.
- **ما هو الخرج من العبارة الآتية:**
- **cout** << “\n **\n ***\n ****\n”;

54 C++ Programming (1)